https://doi.org/10.1051/epjconf/202024502036

Geant4 performance optimization in the ATLAS experiment

Miha Muškinja[1,*], *John Derek* Chapman[3,], and *Heather* Gray[1,2,]
on behalf of the ATLAS Collaboration

[1]Lawrence Berkeley National Laboratory
[2]UC Berkeley, Berkeley, CA
[3]University of Cambridge

Abstract. Software improvements in the ATLAS Geant4-based simulation are critical to keep up with evolving hardware and increasing luminosity. Geant4 simulation currently accounts for about 50% of CPU consumption in ATLAS and it is expected to remain the leading CPU load during Run 4 (HL-LHC upgrade) with an approximately 25% share in the most optimistic computing model. The ATLAS experiment recently developed two algorithms for optimizing Geant4 performance: Neutron Russian Roulette (NRR) and range cuts for electromagnetic processes. The NRR randomly terminates a fraction of low energy neutrons in the simulation and weights energy deposits of the remaining neutrons to maintain physics performance. Low energy neutrons typically undergo many interactions with the detector material and their path becomes uncorrelated with the point of origin. Therefore, the response of neutrons can be efficiently estimated only with a subset of neutrons. Range cuts for electromagnetic processes exploit a built-in feature of Geant4 and terminate low energy electrons that originate from physics processes including conversions, the photoelectric effect, and Compton scattering. Both algorithms were tuned to maintain physics performance in ATLAS and together they bring about a 20% speed-up of the ATLAS Geant4 simulation. Additional ideas for improvements, currently under investigation, will also be discussed in this paper. Lastly, this paper presents how the ATLAS experiment utilizes software packages such as Intel's VTune to identify and resolve hot-spots in simulation.

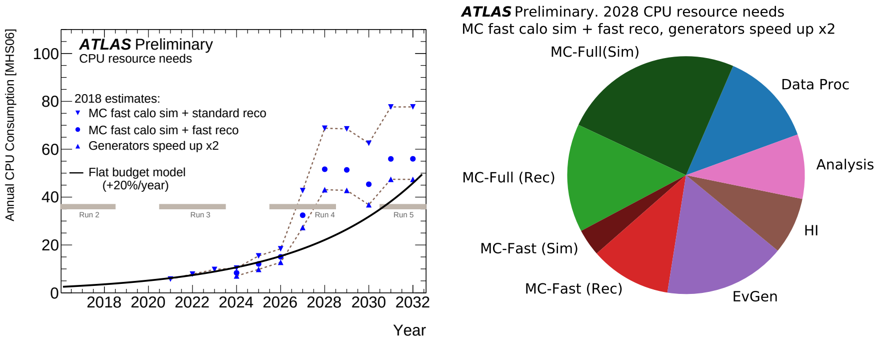
1 Introduction

Detector simulation is an essential tool for data analysis and the interpretation of physics measurements in High Energy Physics (HEP) experiments such as ATLAS. Firstly, physics processes under study are generated with dedicated Monte Carlo (MC) event generator software packages which pseudo-randomly provide final state particles in the detector. Next, the passage of the generated primary particles through the detector material and its magnetic field is simulated to obtain the detector response and to reconstruct the event. Good physics performance of the simulation is desired so that the MC generated events resemble data as closely as possible, however, high precision comes at a cost of high computing time. In the ATLAS experiment, the Geant4 simulation toolkit [1] is used for the most precise simulation. Geant4 is a software package that provides detector geometry and material description tools,

[*] e-mail: mihamuskinja@lbl.gov

and incorporates a number of models to precisely simulate electromagnetic and hadronic interactions of particles with the detector material. When appropriate, various fast simulation methods (e.g. fast calorimeter simulation) can be used to save computing time with some penalty on physics accuracy.

For the MC production used for Run 2 data analysis, ATLAS spent around 50% of CPU resources on Geant4 simulation because the majority of background samples were simulated with Geant4. For Run 3 and beyond, this will no longer be feasible because of the increased luminosity and prohibitively large MC campaigns—that would exceed the available CPU allocation—would be needed. Fast simulation techniques are being developed and it is expected that the majority of MC events used in physics analysis will be simulated using these techniques. However, the need for Genat4 simulation will remain because a precise detector simulation is crucial to develop and tune various fast simulations and the precision of fast simulations will not be sufficient for some high precision measurements, specifically measurements affected by calorimeter-quantities like shower shapes or jet substructure. Figure 1a shows the estimated CPU resources needed for the years 2018 to 2032 for both data and simulation processing. It is evident that there is some discrepancy between the resources expected to be available and even the most optimistic scenario (Fast Calo sim + fast reco + Generators speed ×2). Further, figure 1b shows the fraction of CPU resources needed in 2028 at the end of Run 4 for different processing workflows. Even though most of MC events will be simulated by fast simulations, Geant4 simulation will be the largest CPU consumer with a fraction of about 25%.



(a)

(b)

Figure 1: (a) Estimated CPU resources (in MHS06) needed for the years 2018 to 2032 for both data and simulation processing. The blue points show the improvements possible in three different scenarios, which require significant development work: (1) top curve with fast calo sim used for 75% of the Monte Carlo simulation; (2) middle curve using in addition a faster version of reconstruction; (3) bottom curve, where the time spent in event generation is halved. The solid line shows the amount of resources expected to be available if a flat funding scenario. (b) Fraction of CPU resources needed in 2028 at the end of Run 4 for different processing workflows. The 'MC-Full' section in green is related to the fraction of time spent on the full Atlas Geant4 simulation and divided in a simulation part '(Sim)' for the Geant4 simulation and a reconstruction part '(Rec)' accounting the time spent reconstructing the events. Similarly, the 'MC-Fast' section in red shows this distribution for the time spent running the Fast Calo simulation. Values assume scenario (3)– a faster version of reconstruction and event generation speed-up by a factor of two. Figures taken from ref. [2].

Considering the distribution of CPU resources among data processing workflows in Run 3 and Run 4, outlined in the previous paragraph, it is evident that optimizations of Geant4 performance play a crucial role in approaching the CPU needs issue portrayed in figure 1a. This work introduces two algorithms designed to optimize Geant4 CPU consumption while keeping the physics performance compatible to the 'full' simulation: Neutron Russian Roulette and range cuts for electromagnetic processes. The two optimization strategies together bring a speedup of about 20% in ATLAS Geant4 simulation. The physics output with these algorithms has been validated and stays compatible with the full simulation. Details on the two algorithms are given in section 2.

2 Geant4 Optimization Algorithms in ATLAS

For Run 2 MC production, ATLAS Geant4 simulation already included some speed-up algorithms as outlined in ref. [3]. Namely, ATLAS used 'frozen shower libraries' in forward calorimeters, ClassicalRK4 stepper instead of NystromRK4[1], and 250 ns lifetime cut for neutrons. Together these bring a speed-up of about 35%. All further optimization performance algorithms described in this work are evaluated with respect to the 'Run 2 production' configuration which includes the above mentioned optimizations. In general, simulation time is determined by the particle flux and the geometry complexity and goes into calculating distances between the transported particle and detector components and into providing hit collision. Simulation time can be reduced by either processing fewer steps[2] or reducing the time it takes to process a single step.

Several physics processes have very high cross sections at low energies (e.g. bremsstrahlung, ionisation, electron-positron pair by muons) and it is therefore necessary to implement a production cut so that all particles below it are not generated. Geant4 offers a solution with the so-called 'range cuts', where it is possible to specify an energy threshold below which secondary particles are not created and their energy is immediately deposited at the end of the previous step. Range cuts are passed in units of distance and are converted to an energy threshold internally by Geant4. Further, it is possible to specify range cuts for each material-volume pair separately for gamma, e-, e+, and protons. ATLAS tunes these values depending on the material distribution so that the effect on physics performance is negligible. Considering the example of the ATLAS LAr calorimeter, secondaries created in the passive material (Lead) may not be energetic enough to reach the active material (LAr). In this case physics output will not change substantially if the secondaries are not simulated and their energy is immediately deposited in Lead. Specifically, ATLAS uses a range cut of 100 μm in Lead plates of LAr, which are 2 mm to 3 mm thick.

With the default Geant4 configuration range cuts are only applied to few processes: ionisation, bremsstrahlung, and electron-positron production by muons. In this work we studied the effect of extending the range cuts to compton, photo-electric, and conversion processes (these are major processes where gammas create secondary electrons) by using the energy threshold values already tuned previously for each material-volume pair. We find that the speed-up gained from using range cuts for these processes amounts to about 8% of total simulation time. An effect on the distribution of the initial kinetic energy of electrons in simulation is shown in figure 2a. Vertical gray dashed lines indicate range cut values for some materials where the effect is most noticeable. In total, the amount of simulated (low energy)

[1] ClassicalRK4 is an ATLAS-specific stepper, which NystromRK4 was originally based on.

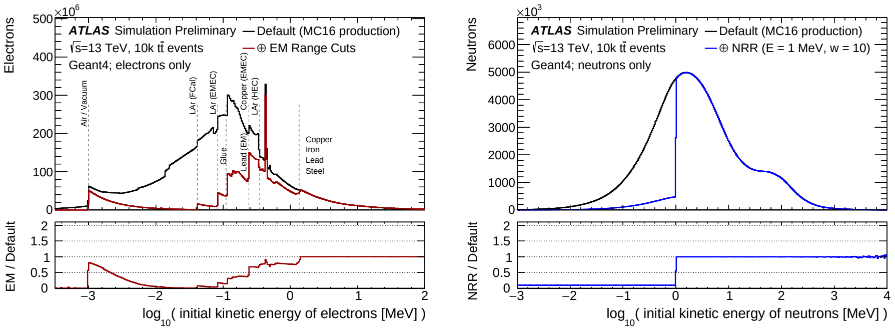
[2] Steps are basic units of Geant4 simulation. Particles are transported in steps where the step can either be caused by a physics process or it can indicate that the particle has reached a material boundary via transportation in the magnetic field.

electrons is decreased by roughly 60%. Thorough physics validation studies show that using these additional range cuts has no negative impact on physics performance.

The second studied optimization algorithm is Neutron Russian Roulette which is based on the fact that neutrons produce many steps (>100) in calorimeters and their trajectory becomes only loosely correlated to the point of origin. Neutron Russian Roulette is an algorithm where neutrons below some energy threshold (E_{th}) are randomly discarded with a probability of $((w-1)/w)$ and the surviving neutrons are weighted with a weight of w. For example, a weight of 10 would indicate that 90% of neutrons below threshold energy are randomly discarded. Technically, weighting is implemented with the SetWeight method of the G4Track object and Geant4 automatically ensures that this weight is propagated to secondary particles created by the initial particle. This weight is then used to scale energy deposits in sensitive detectors by this amount to conserve energy. Practically, the algorithm uses only a fraction of neutrons to predict the energy deposition distribution of all neutrons in the detector. To avoid large weights, already weighted neutrons are not subject to further rouletting. The distribution of the neutron initial kinetic energy along with the effect of a Neutron Roussian Roulette with a threshold of 1 MeV and weight of 10 is presented in figure 2b. The distribution peaks at about 1 MeV and the majority of neutrons are within 0.1 M to 10 MeV range. Further, figure 3a shows the average number of steps that a neutron undertakes in the ATLAS Geant4 simulation. The average number of steps peaks at around 1 MeV where neutrons take about 100 steps before depositing all their energy and the number of steps drops sharply by 50% for an initial energy of 5 MeV. Considering these distributions, ATLAS tested various configurations of Neutron Russian Roulette with rigorous physics validation tests and it was determined that a Neutron Russian Roulette with a weight of 10 is valid up to an energy threshold of 2 MeV. For this configuration, the speed-up is roughly 10% in total simulation time.

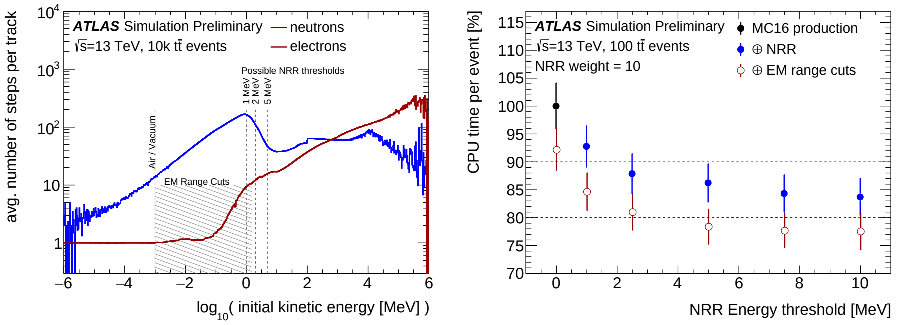
A summary of the speed-up gained from range cuts for electromagnetic processes and the Neutron Russian Roulette algorithm for various energy thresholds and a weight of 10 is presented in figure 3b. Satisfactory physics performance was achieved with range cuts plus the Neutron Russian Roulette with a threshold of 2 MeV, which corresponds roughly to a 20% speed-up in total simulation time. Further, the combined effect on the number of Geant4 steps in various detector volumes is presented in figure 4. Most steps occur in the electromagnetic end-cap calorimeter, where the reduction in the number of steps from the two algorithms is also the most substantial. The second 'heaviest' volume is the electromagnetic barrel calorimeter, followed by 'ID services' and TRT. The number of Geant4 steps in TRT stays the same after applying the two optimization algorithms because none of them directly affects gammas which are the most abundant particles in TRT.

Further optimization performance methods being tested for the ATLAS Geant4 simulation include general software benchmarking and improvements with tools such as Intel's VTune, geometry optimizations and the use of the VecGeom package [5] for geometry calculations. Profiling with VTune shows that there are no major hot-spots and the CPU is spread evenly across many modules which makes it difficult to achieve substantial performance gains with code optimization. Replacing few trigonometrical functions with faster versions (that have lower precision) brought about a 2% speed-up in total simulation time while keeping the physics output sufficiently accurate. Further, early studies show that using VecGeom for geometry calculations could bring a speed-up of up to 4%. Performance gains from VecGeom are mostly limited by the electromagnetic end-cap calorimeter volume, which is a custom solid implementation and not a Geant4 solid-type and therefore gets no benefits from VecGeom. Moreover, geometry studies show that a more efficient description of the bounding volume of the electromagnetic end-cap calorimeter could increase the overall CPU performance by a few percent.



(a) (b)

Figure 2: (a) Distribution of the initial kinetic energy of electrons in the ATLAS Geant4 simulation. The black curve shows the distribution for the default setup (MC16 production) and the red curve shows the distribution for the default setup plus the added Range Cuts for electromagnetic Geant4 processes ('conv', 'phot', 'compt'). Vertical gray dashed lines indicate range cut values for some materials and the right-most dashed line indicates an area with multiple Range Cuts in close proximity for various metals. (b) Distribution of the initial kinetic energy of neutrons in the ATLAS Geant4 simulation. The black curve shows the distribution for the default setup (MC16 production) and the blue curve shows the distribution for the default setup plus the Neutron Russian roulette (NRR) algorithm with energy threshold 1 MeV and weight 10. Figures taken from ref. [4].



(a) (b)

Figure 3: (a) Average number of Geant4 steps per track as a function of the initial kinetic energy in the default setup (MC16 production). Vertical lines indicate the potential energy threshold for the NRR algorithm and the hatched box indicates where Range Cuts have the largest effect. (b) CPU time per event for various thresholds of the NRR algorithm with respect to the average MC16 value (black dot) with or without the EM range cuts. Error bars indicate the RMS of CPU time for the simulated events. Figures taken from ref. [4].

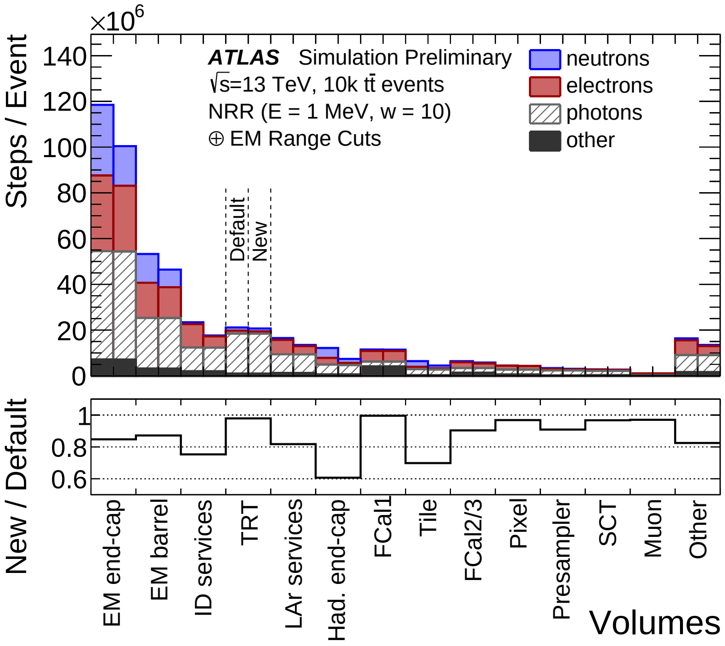


Figure 4: Number of Geant4 steps per event for various ATLAS detector volumes. Left column in each section represents the default setup and the right column represents the default setup with both improvements (NRR and EM range cuts). 'FCal1' includes the first (electromagnetic) module of the forward calorimeter and 'FCal2/3' includes the subsequent two hadronic modules. 'ID services' includes ID services and the beam pipe. 'LAr services' includes LAr services and LAr cryostats. 'Other' includes all other particles and all other volumes that are simulated. Figure taken from ref. [4].

3 Summary

Two performance optimization algorithms were developed to speed-up ATLAS Geant4 simulation: Neutron Russian Roulette, which yields a speed-up of roughly 10% and range cuts for electromagnetic processes which bring an additional 8% speed-up. Furthermore, general software improvements resulting from VTune profiling achieved a speed-up of roughly 2%. All these optimizations were tested on top of the optimizations already in place for Run 2 MC production and the physics performance of the simulation stays compatible with the previous simulation configuration. In view for Run 3, other optimizations such as the use of VecGeom, geometry simplifications, and a Photon Russian Roulette are being tested. The already achieved 20% speed-up together, with potential further speed-ups, will play a crucial role in solving the CPU shortage that ATLAS expects to have at the end of Run 3 and at the beginning of Run 4.

References

[1] J. Allison et al., Nucl. Instrum. Meth. **A835**, 186 (2016)

[2] The ATLAS Collaboration, *Computing and software public results*, `https://twiki.cern.ch/twiki/bin/view/AtlasPublic/ComputingandSoftwarePublicResults`

[3] J. Apostolakis et al. (HEP Software Foundation) (2018), `1803.04165`

[4] The ATLAS Collaboration, *Atlas geant4 optimization performance*, `https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PLOTS/SIM-2019-001/`

[5] J. Apostolakis et al., J. Phys. Conf. Ser. **608**, 012023 (2015)